



---

# DICAS PARA A CONFIGURAÇÃO DE REDES NEURAIS

Por  
Eugênio Silva  
Anderson Canêdo de Oliveira

## 1 – Introdução

Um dos problemas enfrentados no treinamento de redes neurais tipo *MLP* com treinamento *backpropagation* diz respeito à definição de seus parâmetros. A seleção dos parâmetros de treinamento do algoritmo é um processo tão pouco compreendido que é muitas vezes chamado de “magia negra”. Pequenas diferenças nestes parâmetros podem levar a grandes diferenças tanto no tempo de treinamento como na generalização obtida.

Existem inúmeras pesquisas no sentido de encontrar uma fórmula “mágica” que determine a configuração ideal de uma rede neural para um dado problema. No entanto, até agora o que se tem são apenas sugestões que estão fundamentadas em experiências vividas por especialistas no assunto e no bom senso.

Deve-se ter em mente que é preciso obter um modelo que não seja muito rígido a ponto de não modelar fielmente os dados, mas que também não seja excessivamente flexível a ponto de modelar também o ruído presente nos dados. A idéia é que a rede responda de acordo com as características presentes nos dados de entrada e não exatamente igual aos dados de entrada.

O texto a seguir tem como objetivo auxiliar o leitor na tarefa de selecionar adequadamente, no PRW, os parâmetros para a criação e treinamento de uma rede neural *MLP backpropagation* que consiga uma boa generalização em um determinado problema. Vale ressaltar que o conteúdo apresentado aqui deve ser interpretado como sugestão e nunca como um conjunto de regras que devem ser seguidas com rigor.

## 2 – Número de Camadas Escondidas

A utilização de um grande número de camadas escondidas não é recomendado. Cada vez que o erro médio durante o treinamento é utilizado para atualizar os pesos das sinapses da camada imediatamente anterior, ele se torna menos útil ou preciso. A única camada que tem uma noção precisa do erro cometido pela rede é a camada de saída. A última camada escondida recebe uma estimativa sobre o erro. A penúltima camada escondida recebe uma estimativa da estimativa, e assim por diante. Testes empíricos com a rede neural *MLP backpropagation* não demonstram vantagem significativa no uso de duas camadas escondidas ao invés de uma para problemas

menores. Por isso, para a grande maioria dos problemas utiliza-se apenas uma camada escondida quando muito duas e não mais que isso.

Para a resolução de problemas de classificação uma rede neural com uma camada escondida é mais que suficiente. Os problemas de estimação de função também podem ser resolvidos utilizando-se apenas uma camada mas, às vezes, pode ser necessário a utilização de duas camadas escondidas.

### 3 – Número de Neurônios na Camada Escondida

Com relação ao número de neurônios nas camadas escondidas, este é geralmente definido EMPIRICAMENTE. Deve-se ter cuidado para não utilizar nem unidades demais, o que pode levar a rede a memorizar os dados de treinamento (*overfitting*), ao invés de extrair as características gerais que permitirão a generalização, nem um número muito pequeno, que pode forçar a rede a gastar tempo em excesso tentando encontrar uma representação ótima. Devido a estas dificuldades é recomendado manter o número de neurônios escondidos baixo, mas não tão baixo quanto o estritamente necessário.

Existem várias propostas de como determinar a quantidade adequada de neurônios nas camadas escondidas de uma rede neural. As mais utilizadas são:

- Definir o número de neurônios em função da dimensão das camadas de entrada e saída da rede. Pode-se definir o número de neurônios na camada escondida como sendo a média aritmética ou ainda como sendo a média geométrica entre tamanho da entrada e da saída da rede.
- Utilizar um número de sinapses dez vezes menor que o número de exemplos disponíveis para treinamento. Se o número de exemplos for muito maior que o número de sinapses, *overfitting* é improvável, no entanto pode ocorrer *underfitting* (a rede não converge durante o seu treinamento).

### 4 – Taxa de Aprendizado

O parâmetro taxa de aprendizado tem grande influência durante o processo de treinamento da rede neural. Uma taxa de aprendizado muito baixa torna o aprendizado da rede muito lento, ao passo que uma taxa de aprendizado muito alta provoca oscilações no treinamento e impede a convergência do processo de aprendizado. Geralmente seu valor varia de 0.1 a 1.0. Alguns softwares disponíveis no mercado possuem este parâmetro adaptativo, por isso a escolha de um valor inicial não constitui um grande problema. No entanto, no PRW o parâmetro taxa de aprendizado é fixo e por isso sugere-se não utilizar um valor muito alto. Uma sugestão seria o valor 0.4.

### 5 – Momentum

A inclusão do termo *momentum* tem por objetivo aumentar a velocidade de treinamento da rede neural e reduzir o perigo de instabilidade. Este termo pode ou não ser utilizado durante o

treinamento e seu valor varia de 0.0 (não utilização) a 1.0. O valor recomendado para o termo *momentum* é 0.3.

## 6 – Dinâmica de Treinamento

### 6.1 – Por padrão (on-line ou incremental)

No treinamento por padrão, os pesos são atualizados após a apresentação de cada exemplo de treinamento. Nesta abordagem, a ordem da apresentação dos padrões é importante para a velocidade de aprendizado da rede e, em alguns casos, deve-se reorganizar esta ordem, de forma a acelerar o treinamento.

A dinâmica de treinamento por padrão é ESTÁVEL se a taxa de aprendizado for PEQUENA (é aconselhável reduzir progressivamente esta taxa durante o treinamento). Quando taxas elevadas são utilizadas, a rede geralmente se torna INSTÁVEL. A abordagem por padrão é geralmente mais rápida, principalmente se o conjunto de treinamento for grande e redundante. Uma outra vantagem desta técnica é que ela requer menos memória.

### 6.2 – Por ciclo (batch ou epoch)

Na abordagem por ciclo os pesos são atualizados após todos os exemplos de treinamento terem sido apresentados. Esta técnica é geralmente mais ESTÁVEL e o treinamento é menos influenciado pela ordem de apresentação dos padrões, mas ela pode ser lenta se o conjunto de treinamento for grande e redundante. Uma outra desvantagem é que ela requer mais memória. Comparativamente com o treinamento incremental, o treinamento por *batch* é geralmente mais estável embora mais lento.

## 7 – Critérios de Parada do Treinamento

Existem vários métodos para a determinação do momento em que o treinamento de uma rede neural deve ser encerrado. Uma boa determinação destes critérios é fundamental para um bom treinamento e conseqüentemente uma boa generalização. Os critérios de parada mais utilizados são:

- **Número de ciclos**  
Define o número de ciclos de treinamento, ou seja, o número de vezes em que o conjunto de treinamento é apresentado à rede. Um número excessivo de ciclos pode levar a rede à perda do poder de generalização (*overfitting*). Por outro lado, com um pequeno número de ciclos a rede pode não chegar ao seu melhor desempenho (*underfitting*). Sugere-se um valor entre 500 e 3000 ciclos de treinamento.
- **Erro**  
Consiste em encerrar o treinamento após o erro médio quadrático ficar abaixo de um valor *a* pré-definido. Vale lembrar que um erro médio quadrático muito pequeno não implica necessariamente numa boa generalização (i.e. bom desempenho com dados não vistos).

anteriormente). Este valor depende muito do problema. Uma sugestão é estabelecer um valor de 0.01 no primeiro treinamento e depois ajustá-lo em função do resultado.

- **Combinação dos Métodos Anteriores**  
Também se pode estipular como método de parada uma combinação dos métodos citados acima. Desta forma, o treinamento é encerrado quando qualquer um dos critérios acima é atendido.
- **Validação (Best Model)**  
Na técnica de parada pela validação, o treinamento é interrompido a cada  $x$  ciclos e é realizada uma estimação de erro da rede sobre o conjunto de dados de teste. A partir do momento em que o erro medido no conjunto de teste apresentar crescimento, o treinamento é encerrado. O que se deseja com esta técnica é descobrir o momento exato em que a rede começa a perder generalização.

## 8 – Consideração Final

Mesmo tomando todos os cuidados citados aqui para a configuração e o treinamento de redes neurais, pode acontecer de o resultado final não ser satisfatório. Quando isso acontece recomenda-se realizar novamente o treinamento da rede mantendo os parâmetros inalterados. Se esta prática não levar a um bom resultado deve-se realizar pequenas alterações nos parâmetros. Num primeiro momento deve-se alterar apenas os parâmetros de treinamento e executar novos treinamentos. Se ainda assim não houver melhora nos resultados deve-se então partir para a alteração dos parâmetros de configuração da rede e mais uma vez executar novos treinamentos. Se todas estas tentativas não surtirem efeito, provavelmente o problema se encontra nos dados de treinamento. Estes podem não estar caracterizando bem o problema que se pretende resolver. Persistindo os maus resultados é melhor esquecer e partir para outra técnica na tentativa de resolver o problema.

## 9 – Para Saber Mais

- 1 – Bishop, C. M.; *Neural Networks for Pattern Recognition*, Oxford University Press; 1995.
- 2 – Braga, A. P., Carvalho, A. P. L. F., Ludemir, T. B.; *Fundamentos de Redes Neurais Artificiais*, 11ª Escola de Computação; Rio de Janeiro, de 20 a 24 de julho de 1998.
- 3 – Haykin, S.; *Neural Networks: A Comprehensive Foundation*, Prentice Hall; 1999.
- 4 – Zurada, J. M.; *Introduction to Artificial Neural Systems*, PWS Publishing; 1992.
- 5 – <http://www.faqs.org/faqs/ai-faq/neural-nets/>